

High Precision Accelerator Cavity Design Using the Parallel Eigensolver Omega3P^{*}

Yong Sun[†], Nate Folwell[‡], Zenghai Li[§] and Gene H. Golub^{**}

Abstract. In this paper, we report our work on high precision accelerator cavity design using the parallel eigensolver Omega3P. Omega3P is the software implementation of the Filtering algorithm [1] for solving large generalized eigenproblems. With Omega3P, we have successfully solved many accelerator cavity design problems with high precision and good parallel performance.

1 Introduction

Generalized eigenproblems of the form $\mathbf{K} \mathbf{x} = \lambda \mathbf{M} \mathbf{x}$ arise naturally in accelerator cavity design problems. These eigenproblems have the following properties: the stiffness matrix \mathbf{K} and the mass matrix \mathbf{M} are real symmetric, and matrix \mathbf{M} is positive definite. Usually a number of interior eigenpairs of $\mathbf{K} \mathbf{x} = \lambda \mathbf{M} \mathbf{x}$ are desired.

The matrix eigenproblems are generated from finite element methods, and therefore the matrices are very sparse, often less than 20 non-zero elements per row for linear finite elements. To achieve the accuracy required in accelerator cavity designs, the matrices are often the size of millions by millions.

A fast, accurate and parallel eigensolver is highly desirable for the matrix eigenproblems. The parallel eigensolver Omega3P and its theoretical foundation, the Filtering algorithm, are jointly developed by SCCM and SLAC in Stanford University for this purpose.

The rest of the paper is organized as follows. In section 2, we study the numerical properties of the eigenproblems in accelerator cavity designs. In section 3, we describe the Filtering algorithm, which serves as the theoretical foundation of Omega3P. In section 4, we describe the software implementation and the performance of Omega3P, and in section 5, we make concluding remarks.

2 Numerical Properties of Our Eigenproblems

Given the geometry of a conductive cavity Ω with its material properties, permittivity ϵ and permeability μ , we desire to find the eigenfrequencies ω and the corresponding field distributions \mathbf{E} that satisfy the curl-curl formulation of Maxwell's equation in Ω .

$$\begin{aligned} \nabla \times (\mu^{-1} \nabla \times \mathbf{E}) &= \omega^2 \epsilon \mathbf{E} & \text{in } \Omega \\ \hat{n} \times \mathbf{E} &= 0 & \text{on } \partial\Omega \end{aligned} \tag{2.1}.$$

For problems that we consider, the dielectric materials are lossless and uniformly distributed in Ω , and therefore the eigenvalues of (2.1) are $\omega^2 \epsilon \mu$. It is apparent that all eigenvalues of (2.1) are nonnegative, and their corresponding eigenvectors are real [2]. It can be shown that (2.1) has infinite number of solutions when ω is zero. It is also clear that the largest ω of (2.1) is infinity. In practice, only the first few of non-zero eigenvalues are of interest for cavity design purposes.

The variational representation of the curl-curl Maxwell's equation (2.1) is:

^{*} Work supported by the U.S. DOE Contract No. DE-AC03-76SF00515

[†] SCCM, Stanford University, Stanford, CA 94305, USA. ysun@sccm.stanford.edu

[‡] SLAC, Stanford University, Stanford, CA 94309, USA. folwell@slac.stanford.edu

[§] SLAC, Stanford University, Stanford, CA 94309, USA. lizh@slac.stanford.edu

^{**} SCCM, Stanford University, Stanford, CA 94305, USA. golub@sccm.stanford.edu

Find (ω, \mathbf{E}) such that

$$(\nabla \times \mathbf{F}, \mu^{-1} \nabla \times \mathbf{E}) = \omega^2 (\mathbf{F}, \varepsilon \mathbf{E}) \quad \forall \mathbf{F} \in H(\text{curl}, \Omega, \partial\Omega) \quad (2.2).$$

$$\text{Where } \left\{ \begin{array}{l} H(\text{curl}, \Omega, \partial\Omega) = \{ \mathbf{E} \in H(\text{curl}, \Omega) : \hat{n} \times \mathbf{E} = 0 \text{ on } \partial\Omega \} \\ H(\text{curl}, \Omega) = \{ \mathbf{E} \in L^2(\Omega) : \nabla \times \mathbf{E} \in L^2(\Omega) \} \\ L^2(\Omega) = \left\{ \mathbf{E} : \int_{\Omega} |\mathbf{E}|^2 d\Omega < \infty \right\} \\ (\mathbf{F}, \mathbf{E}) = \int_{\Omega} \mathbf{F}^* \cdot \mathbf{E} d\Omega \quad \forall \mathbf{E}, \mathbf{F} \in L^2(\Omega) \end{array} \right.$$

The discretization of finite element equation (2.2) leads to the following matrix eigenproblem:

$$\mathbf{K} \mathbf{x} = \lambda \mathbf{M} \mathbf{x} \quad (2.3).$$

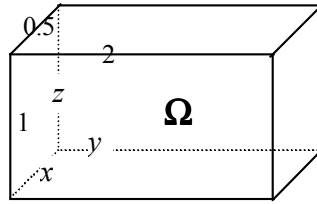
In the matrix eigenproblem (2.3), stiffness matrix \mathbf{K} and mass matrix \mathbf{M} are real symmetric and of the same sparsity pattern, and matrix \mathbf{M} is positive definite. Without finite element discretization errors, matrix \mathbf{K} shall be positive semi-definite. In practice, due to finite element discretization errors, matrix \mathbf{K} is near positive semi-definite.

Eigenvalues of the matrix eigenproblem (2.3) are the discrete version of the eigenvalues of the Maxwell's equation (2.1), and therefore have the following properties.

1. A number of the eigenvalues of the matrix eigenproblem (2.3) are very close to zero.
2. Excluding those “zero” eigenvalues, the first few smallest eigenvalues and their corresponding eigenvectors are wanted. Including the “zero” eigenvalues, the first few “nonzero” interior eigenvalues and their corresponding eigenvectors are wanted.
3. The largest eigenvalues of (2.3) are much larger than the eigenvalues of interest.

It is worth noting that these properties are quite general in cavity designs and other structural simulations.

We shall now study these characteristic properties through our model problem. For simplicity, in our model problem, Ω is a 0.5m x 2m x 1m rectangular box, with vacuum inside ($\varepsilon = \mu = 1$).



The problem has analytical solutions for its eigenvalues given in (2.4), where c is the speed of light; m, n, p are integers. The smallest nonzero eigenvalue λ_0 is 12.33, and it corresponds to $(m, n, p) = (0, 1, 1)$.

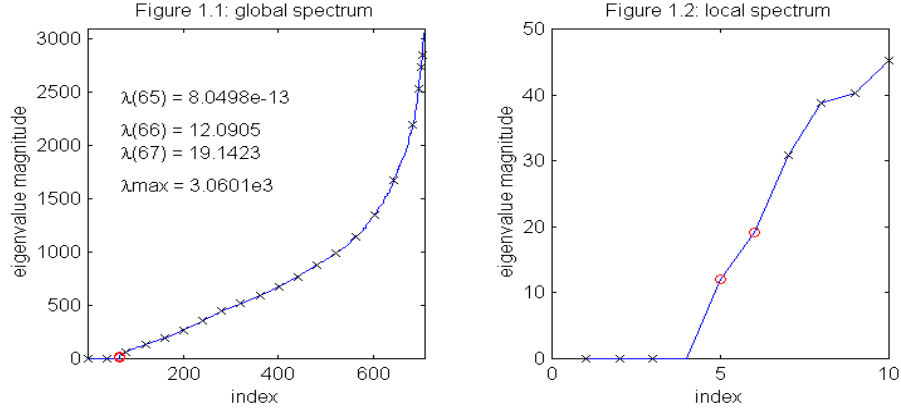
$$\lambda_{mnp} = \pi^2 \left[\left(\frac{m}{x} \right)^2 + \left(\frac{n}{y} \right)^2 + \left(\frac{p}{z} \right)^2 \right], \text{ where } m, n, p = 0, 1, 2, \dots \quad (2.4).$$

$$\lambda_0 = 12.33$$

When the edge based linear finite element method is applied on a rather coarse tetrahedral mesh (233 mesh points) for Ω , the dimension of the resulting matrices, i.e., matrix \mathbf{K} and \mathbf{M} , is 704, and there are about 13 nonzero elements per row for each matrix. We will use this model problem throughout the rest of the paper.

The following spectrum plot of the model problem exhibits the characteristic spectrum properties of the structural simulation problems.

Figure 1: Spectrum distribution of the model problem



It is this spectrum distribution that poses significant difficulty to many existing eigenvalue algorithms especially when the problem size is big. The challenge was the motivation of the Filtering algorithm.

3 The Filtering Algorithm

The Filtering algorithm was developed particularly for solving large generalized eigenproblems from Maxwell's equations.

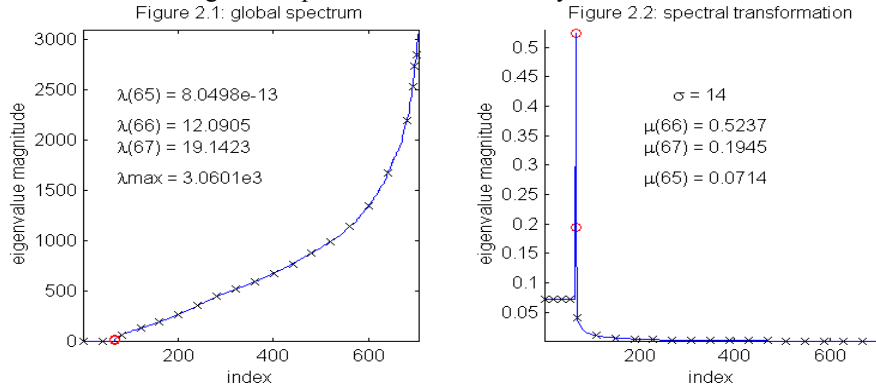
The Filtering algorithm is a hybrid scheme that consists of two techniques: Inexact Shift and Invert Lanczos (**ISIL**) for obtaining eigenvector approximations efficiently; and Jacobi Orthogonal Component Correction (**JOCC**) algorithm for refining the eigenvector approximations. The first technique is regarded as the filtering process to get good eigenvector approximations. The second technique is only required when the eigenvector approximations from the **ISIL** filtering process is not accurate enough.

Mathematically, Shift and Invert Lanczos (**SIL**) algorithm is ideal for solving interior eigenproblems. It converts interior eigenvalues closest to the shift σ into the largest ones of new eigenproblems, and the largest eigenvalues are well separated from other eigenvalues. The spectrum transformation is written as follows.

$$\mathbf{K} \mathbf{x} = \lambda \mathbf{M} \mathbf{x} \Rightarrow \mathbf{M} (\mathbf{K} - \sigma \mathbf{M})^{-1} \mathbf{M} \mathbf{x} = \mu \mathbf{M} \mathbf{x}$$

$$\lambda \Rightarrow \mu = \frac{1}{\lambda - \sigma} \quad (3.1).$$

Figure 2: Spectral transformation by Shift and Invert

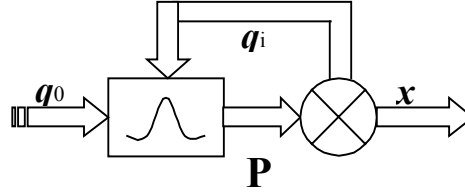


The spectral transformation (3.1) amplifies the magnitude of eigenvalues close to σ , and dampens others away from σ . When the shift $\sigma = 14$, which is chosen to be close to the first nonzero eigenvalue λ_0 in the analytical solution (2.4) of the model problem, the transformed spectrum of the model problem is shown in Figure 2.

The **SIL** algorithm has a fatal drawback. At each iteration step, a shifted linear system, $(\mathbf{K} - \sigma \mathbf{M}) \mathbf{x} = \mathbf{b}$, needs to be solved accurately. Unfortunately the shifted matrix is usually large and ill conditioned in our problems, and therefore very difficult to solve.

The reason why the **SIL** algorithm is so effective is that the spectral transformation from the Shift Invert process makes the eigenvalues of interest distinctive from other eigenvalues. The spectral transformation can be regarded a band pass filter that keeps the eigenvalues of interest. In fact, all other Krylov subspace methods obey the filter model shown in Figure 3.

Figure 3: Filter model for Krylov subspace methods



It is shown [1] that for eigenproblems arising from Maxwell's equations, a filter similar to the filter in the **SIL** algorithm can be easily obtained by solving the same shifted linear systems inexactly. We call this filter Inexact Shift Invert Lanczos (**ISIL**) filter.

It is also shown [1] that the difference between the **SIL** filter and the **ISIL** filter are bounded by the residual tolerance for solving the shifted linear systems inexactly. This implies that even if the shifted linear systems are solved inexactly with a residual tolerance of 0.01, 99% of the filtering result that the **SIL** filter can produce is already achieved by the **ISIL** filter. Therefore we expect that the **SIL** filter and the **ISIL** filter will give almost identical performance in getting eigenvector approximations.

Solving the shifted linear system inexactly with a residual tolerance only at 0.01 requires significantly less amount of computational work and poses much smaller computational challenge than that in the **SIL** filter where the shifted linear systems are solved exactly. More importantly, when problems are large and the shifted linear systems are ill conditioned, it becomes unrealistic to apply the **SIL** filter, while the **ISIL** filter is much more practical.

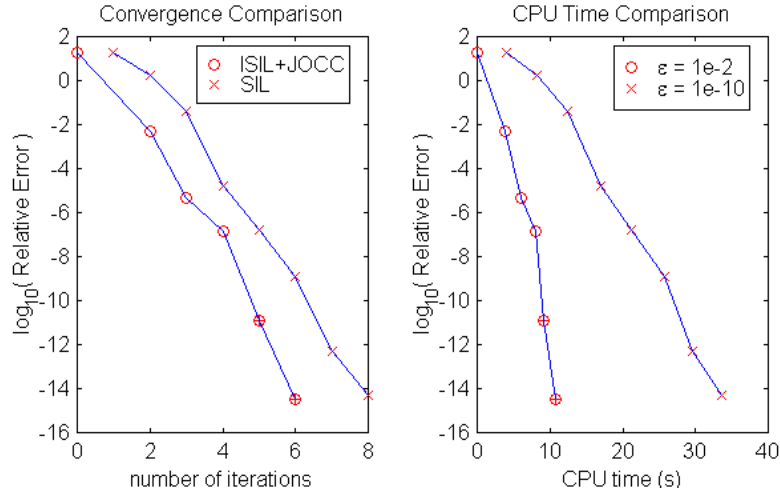
The **ISIL** filter stagnates after a good approximation of the desired eigenvector is reached, because at some point the current eigenvector approximation is so "good" for the **ISIL** filter that additional filtering steps will not help to further refine the approximation.

To obtain a more accurate solution than that from the **ISIL** filter, we use a Newton type scheme, the **JOCC** (Jacobi Orthogonal Component Correction) algorithm, to refine the eigenvector approximation. In the **JOCC** algorithm, similar shifted linear systems need to be solved at each iteration step. However, given a good eigenvector approximation, the **JOCC** refinement process will have super linear convergence even if the shifted linear systems are only solved approximately, and only a very small number (usually 2 or 3) of **JOCC** steps are needed to reach a high precision for the eigenvector.

In the model problem, the shifted matrix, $\mathbf{K} - \sigma \mathbf{M}$, is small enough to be solved accurately. We set the residual tolerance of solving the shifted linear systems in the **SIL** algorithm to be $1e-10$ to achieve roughly the same convergence rate as the case of explicitly inverting the shifted matrix. We set the residual tolerance for solving the shifted linear systems in the **ISIL** filter and the **JOCC** refiner to be both $1e-2$.

The comparison between the two algorithms in terms of both number of outer iterations and computational time is shown in Figure 4.

Figure 4: Comparison between ISIL+JOCC and ESIL



We notice three interesting phenomena in the comparison:

1. The convergence curve starts from 0 for the Filtering algorithm versus 1 for the **SIL** algorithm, because the **ISIL** filter makes use of the last Ritz vector, which is not included into the current Ritz vector space in the **SIL** algorithm.
2. The accuracy of the **ISIL** filter at the 4th interaction is about the same as that of the **SIL** algorithm at the 5th iteration, which verifies that the **ISIL** filter and the **SIL** filter almost have identical filtering effects to obtain a good eigenvector approximation.
3. The last two **JOCC** refinement steps clearly outperform the **SIL** algorithm.

The Filtering algorithm outperforms the **SIL** algorithm even in terms of number of outer iterations. In terms of the CPU time, the computational cost of the Filtering algorithm is about one third that of the **SIL** algorithm.

4 Omega3P and its Applications in Accelerator Cavity Designs

Parallel computing is essential for solving large problems. It has been our main objective in developing the Filtering algorithm. Solving the shifted linear systems of the **ISIL** filter and the **JOCC** refiner is the dominant computational cost. Therefore the parallel implementation of the Filtering algorithm is reduced to parallel preconditioning and matrix-vector multiplication, which are well addressed by many efforts in the parallel computing community.

The fact that the shifted linear systems in the Filtering algorithm are solved only approximately makes it very suitable for parallel computing. Many iterative linear solvers in the public domain are capable of solving such systems approximately. We build Omega3P, the parallel implementation of the Filtering algorithm, on the Aztec library [6].

Omega3P is the complete software package for accelerator cavity modeling on parallel platforms. Besides the eigensolver, which is the parallel implementation of the Filtering algorithm, Omega3P has the following components.

1. DistMesh, for parallel and distributed mesh partitioning and distributing.
2. Parallel finite element formulation for distributed matrix assembling.
3. Parallel post-processing for result output and visualization.

Omega3P works on most of the parallel platforms available today.

1. MPP, massively parallel processor machines, such as CrayT3E and SP2 in NERSC.
2. SMP, symmetric multiple processor machines, such as SUN Enterprise 10000.
3. LINUX PC Cluster, such as the morab cluster in the Advance Computation Department at SLAC.

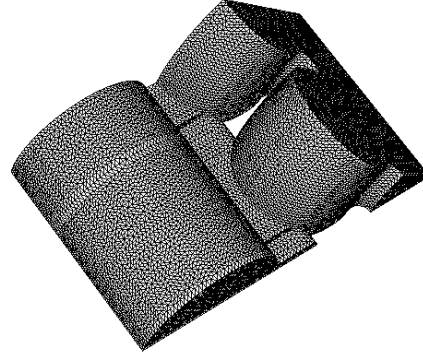
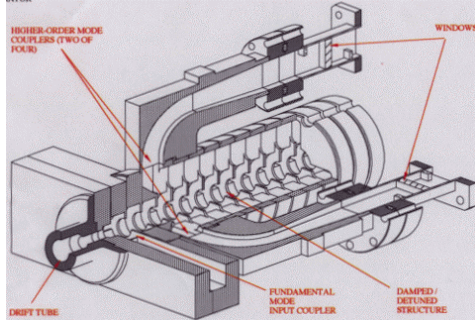
Omega3P has successfully solved large eigenproblems for accelerator cavity design applications. Figure 5 is a realistic DDS (Damped Detuned Structure) design in the NLC (Next Linear Collider) Project at SLAC. It is a complex 3D structure,

consisting of 206 cells, whose fundamental mode needs to be calculated accurately. Figure 5 shows the geometry of the whole DDS and the mesh used in our computation for one of such cells. The resulting matrices are of dimension 1,066,302. The **ISIL** filter alone works well for this problem. With the shifted linear systems being solved to the residual tolerance of $1e-2$, the solver converges to the required accuracy after 4 iteration steps without the help of **JOCC** refiner.

Figure 5: SLAC DDS

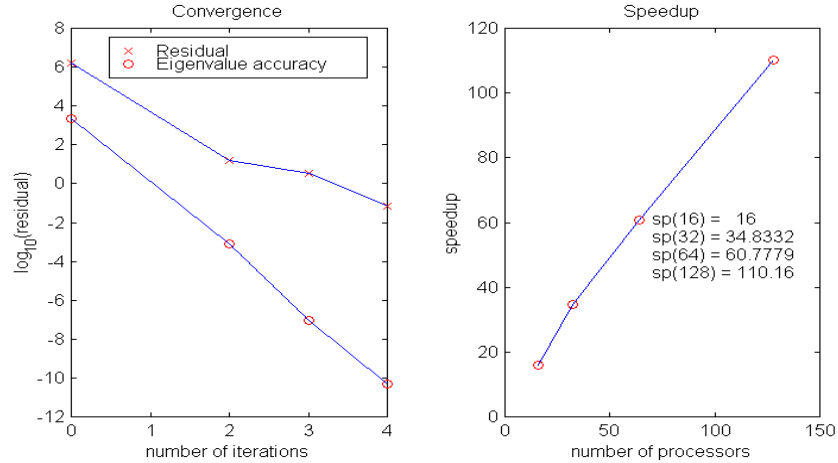
Whole DDS

Mesh for 1/8 of 11/2 RDDS Cells



Omega3P exhibits good speedup and scalability as the number of processors increases. The performance of our solver for the above problem on CrayT3E is given in Figure 6. It takes less than seven minutes for the solver to converge on 128 processors. The speedup of Omega3P on 128 processors is 110.

Figure 6: Parallel performance on CrayT3E



The dimensions of the actual cells were designed using Omega3P. The computation was carried out on a larger matrix eigenproblem whose size is about 5 million by 5 million. The cells have been manufactured, and their experimental measurements agree well with Omega3P numerical predictions.

6 Concluding Remarks

We have developed the Filtering algorithm particularly for solving large eigenproblems arising from accelerator cavity designs. Based on the Filtering algorithm, we have implemented a complete parallel software package Omega3P for accelerator cavity modeling. Omega3P exhibits satisfactory convergence behavior and parallel performance, and it is being widely used by accelerator design scientists in the accelerator design community.

References

- [1] Y. SUN. The Filtering Algorithm for Solving Large Eigenproblems. Ph.D. thesis. SCCM, Stanford University, Stanford, CA, USA. 2002.
- [2] G. H. Golub and C. F. van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, MD 21211, 3rd edition, 1996.
- [3] G. H. Golub and Q. Ye. Inexact inverse iterations for the generalized eigenvalue problems. Sccm-99-02, Stanford University, 1999.
- [4] G. L. G. Sleijpen and H. A. van der Vorst. A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17: 401-425, 1996.
- [5] C. Lanczos. *Applied Analysis*. Englewood Cliffs, N.J., Prentice Hall, 1956.
- [6] S. A. Hutchinson, J. N. Shadid and R. S. Tuminaro. AZTEC user's guide. Technical Report SAND95-1559, Massively parallel computing research laboratory, Sandia National Laboratories, Albuquerque, NM, 1995.